DOCUMENT RESUME

ED 163 906 IR 006 219

AUTHOR Muller, Mervin E.
TITLE Computers as an Instrument for Data Analysis. Technical Report No. 11.
INSTITUTION Wisconsin Univ., Madison.
PUB DATE Aug 68
NOTE 66p.; Parts may not reproduce well due to light print

EDRS PRICE MF-$0.83 HC-$3.50 Plus Postage.
DESCRIPTORS Algorithms; Computer Programs; *Computers; *Data Analysis; Data Processing; Facilities; Futures (of Society); Management Information Systems; *State of the Art Reviews; Systems Analysis; Systems Approach
IDENTIFIERS Computer Software; Digital Transmission Systems; *STATJOB

ABSTRACT

A review of statistical data analysis involving computers as a multi-dimensional problem provides the perspective for consideration of the use of computers in statistical analysis and the problems associated with large data files. An overall description of STATJOB, a particular system for doing statistical data analysis on a digital computer, includes system design goals, the current analytical status, and a flow chart. A brief look at the future of computers in data analysis and a 93-item bibliography conclude the report. (CMV)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

ED163906

Computers as an Instrument for Data Analysis

By Mervin E. Muller

The University of Wisconsin
Computing Center
1210 West Dayton Street
Madison, Wisconsin

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Albert Roberts

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC) AND USERS OF THE ERIC SYSTEM."

Technical Report
Number 11
August, 1968

IR006219

TABLE OF CONTENTS

		PAGE
I.	INTRODUCTION AND PERSPECTIVE	1
	I. 1 Individuals Involved	2
	I. 2 Applications	4
	I. 3 Environment	4
	I. 4 Data Implications	4
	I. 5 Response Time Requirements	5
	I. 6 Applicable Statistical Analysis Techniques	6
	I. 7 Computing Algorithms	7
	I. 8 Computing Facilities - People, Hardware, and Software	7
II.	SOME HISTORICAL CONSIDERATIONS	8
	II. 1 Exponential Growth in the Numbers of Programs	8
	II. 2 Specialization and Generalization	11
	II. 3 Software Development and Its Phases	13
	II. 4 Development of Algorithms and Computing Procedures	18
	II. 5 Some Concluding Remarks Leading to the Present	21
III.	SOME AVAILABLE LARGE SCALE STATISTICAL PACKAGES	22
IV.	IMPLICATIONS OF LARGE DATA FILES	23
	IV. 1 Consequences of Handling Large Data File	24
	IV. 2 Serial, Parallel, and Hybrid Modes of Data	25
	IV. 3 Serial, Random, and Pseudo Random Access to Data	27
	IV. 4 Backup and Error Recovery	29
	IV. 5 Passing Program Over Data	29
	IV. 6 Sampling of Files	30

V. STATJOB 31
V.1 Global Systems View of Data Analysis 31
V.2 System Design Goals of STATJOB 32
V.3 Implementation Accomplishments in STATJOB 38
V.3.1 Control Processing 40
V.3.2 Input Processing 41
V.3.3 Analysis Processing 41
V.3.4 Output Processing 43
V.4 Summary of Actual Experience 44
V.5 Weaknesses of STATJOB 46
V.6 Future Plans for STATJOB 47

VI. THE FUTURE 49

VII. SUMMARY 53

LIST OF REFERENCES 54

TABLES

Table II.1 A Comparison of Programming Approaches 17
Table II.2 A Comparison of Packaging Approaches 17
Table III.1 Large Scale Statistical Packages Operating in Batch Processing Environment 23
Table V.1 User Goals for a Statistical Data Analysis System 33
Table V.2 Designer Goals for a Statistical Data Analysis System 34
Table V.3 Implementors Goals for a Statistical Data Analysis System 35
Table V.4 Maintainer Goals for a Statistical Data Analysis System 36
Table V.3.1 Summary of Current Analytical Status of STATJOB 45

FIGURES

Figure 1 Data Analysis 3

Figure IV. 1	Consequences of File Size	26
Figure V. 1	A Global Systems View of a Statistical Data Analysis System	37
Figure V. 3. 1	Functional Components of STATJOB	42

I. INTRODUCTION AND PERSPECTIVE

The computer has been compared to the steam engine in the sense that the steam engine provided the power for the Industrial Revolution and the computer is viewed by some as the engine of the Information Revolution. However, in terms of the computer as an instrument for data analysis, I would like to compare the computer to the microscope and the telescope. Much of man's progress has come about by the development of instruments that aid observation and measurement. For the computer to be the generally useful and flexible instrument in data analysis that many of us assume it to be, it must be able to assist in observing and measuring phenomena in the small in the sense of the microscope and in the large in the sense of the telescope. In some respects the great enthusiasm and expectations for the computer in the area of data analysis have been severely threatened by the increasing awareness of the disappointments, cost, effort, and time that have been experienced in trying to have the digital computer serve as an effective instrument for data analysis.

In the early uses of computers for statistical analysis, one usually observed the development of a special purpose computer program for a specific problem, followed by subsequent improvements and generalizations of the program. Usually this type of generalization is useful and pleasing to see. However, I believe another type of generalization in terms of multi-dimensional factors is needed if we are to view in a meaningful perspective the computer as an instrument for data analysis. Hopefully this perspective will make it easier to see and distinguish among:

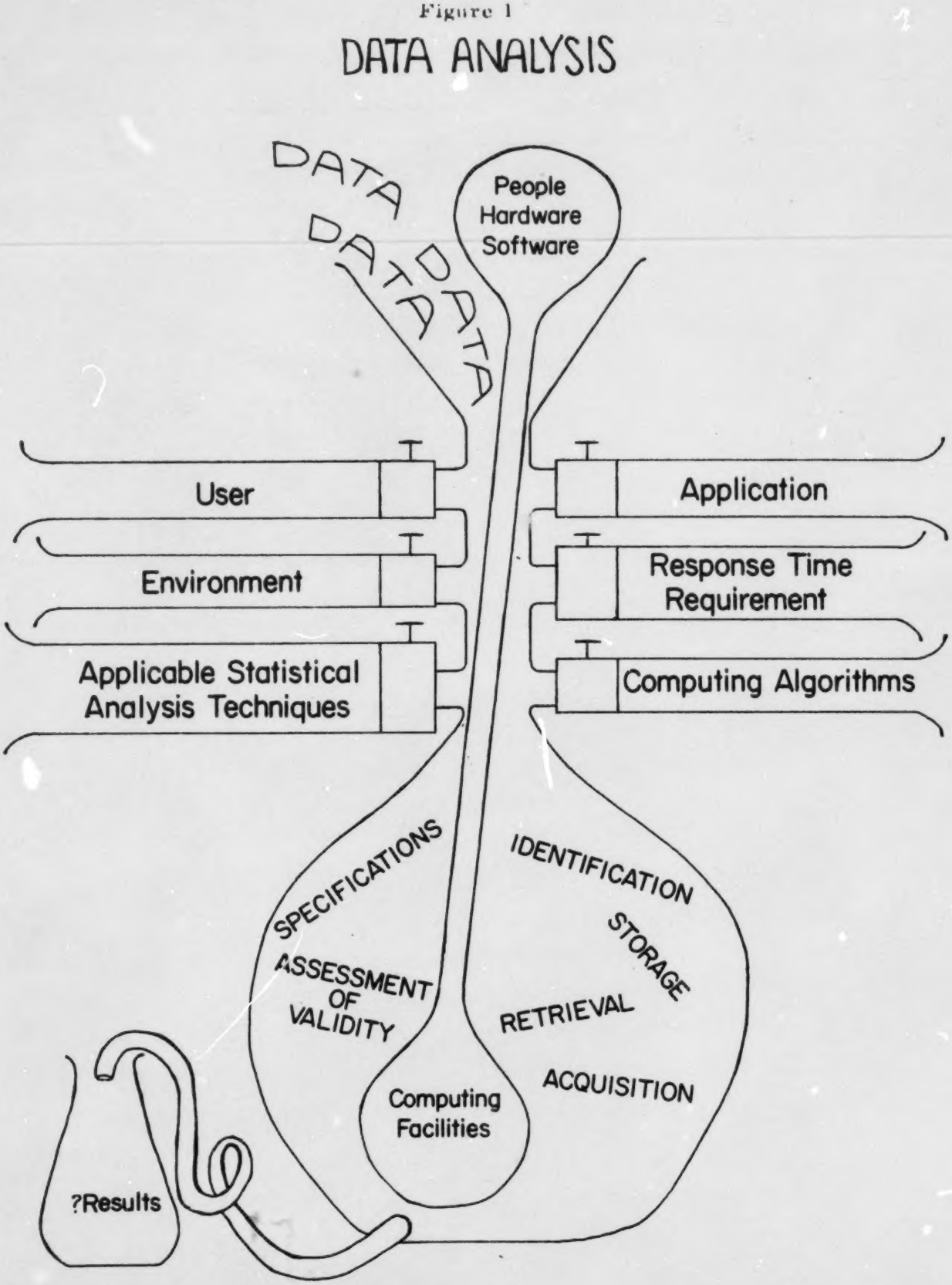
- What is desired
- What is available and accomplished
- What is achievable

In considering a digital computer as an instrument for data analysis, it seems to me to be helpful to think of having an instrument with multiple controls which are highly interactive and influence the quality of the calibration. To use the jargon of the statistician interested in design of experiments, instead of that of the control engineer, we can think of each control knob as a factor. Thus we can think of having a response function which is dependent on the following interacting factors or components: 1) individuals involved, 2) applications, 3) environment, 4) data implications-sources, identification, acquisition, assessment of quality, and storage, 5) response time requirements, 6) applicable statistical analysis techniques, 7) computing algorithms, and 8) computing facilities - people, hardware, and software. These factors are shown pictorially in Figure 1.

I. 1 Individuals Involved

The role of the individuals involved in a data analysis is certainly crucial to what is desired, what ought to be desired, what is achievable, and what is practical. Their knowledge, experience, and understanding of the other factors cited above is often overlooked when trying to determine or judge the use of a computer in data analysis. (Healy in (42) provides a brief but interesting discussion of the need to take into account whether or not the individual is naive or sophisticated with respect to data analysis and/or computing.) Many prior evaluations or designs of computer programs for data analysis were inadequate because they failed to take into account the knowledge of the user. As in the case of using a microscope or telescope, it is asking too much of the designer of the instrument to expect it to be used correctly or effectively in untrained hands. However, often people designing programs for data analysis have failed to identify clearly the class of intended users, or the pitfalls of allowing an untrained user to have access to the computer through the use of these programs. More attention will be given to this point later.

Figure 1


DATA ANALYSIS
DATA
DATA
DATA
People
Hardware
Software
User
Environment
Applicable Statistical
Analysis Techniques
Application
Response Time
Requirement
Computing Algorithms
SPECIFICATIONS
ASSESSMENT
OF
VALIDITY
IDENTIFICATION
STORAGE
RETRIEVAL
ACQUISITION
Computing
Facilities
?Results

I. 2 Applications

Computer use for data analysis can be classified, in a global sense, into one of the following classes depending on whether or not the computer is being applied to aid analysis involving prediction, estimation, explanation, exploration, description, summarization, or control. Within an application class, one can find many diverse subject matter fields. The class and field of application can seriously affect the capability of the computer for data analysis. The availability of a theory or a model to help collect or analyze the data is often application dependent. The field of application is important too; for example, a physical science application, as opposed to a social science application, will often be distinguished by whether or not there is an analytical model to help explain the process. The ability to control the acquisition of data, and the lack of, or even abundance of data are certainly application and field dependent, and in turn effect the contribution to be gained from the computer.

I. 3 Environment

The application and the environment for data acquisition and use can often be very interdependent. The ability to design the data acquisition aspects so as to control or exclude undesired environmental influences certainly affects the ultimate role of the computer. Other aspects of environment dictate whether or not it is possible to have data acquisition equipment present and directly or indirectly connected to the computer. The environment and application also can influence the need to acquire and record data in a serial or parallel mode with regard to the variables of interest. This point will be considered in more detail later in Section IV. 2.

I. 4 Data Implications

The ability to determine the sources of data and the subsequent identification, acquisition, assessment of quality, and storage of data for

subsequent retrieval and/or analysis, in my opinion is dependent upon and affects all of the other factors associated with data analysis. In many respects the ultimate contribution of the computer can be no better than the data it has to process and store, yet it is precisely when there are questionable, missing, or excessive amounts of data that it is so difficult to determine how to use the computer to its full potential. Although we often feel and expect that the computer is needed most to cope with large amounts of data, it is when a user of a computer is in this situation and is faced with analyzing a large amount of data that he becomes most aware of the current limitations of computers and computer programs. This point will be considered in more detail when we dwell on the consequences of large data bases in Section IV.

I. 5 Response Time Requirements

Response time requirements can be interpreted to mean the time in which an analysis must be completed. While this has serious consequences, the aspects of the response time that will be given special attention here relate to those applications and environmental situations where either a computer is controlling the data acquisition or a user is using a computer in an interactive way where it is vital that a response or signal be made within a specified time interval so that the data will not be lost or incorrectly acquired. This type of situation is becoming more prevalent with on-line data acquisition and with interactive time-sharing. In both cases, the response time requirement can put severe limitations on the type of statistical techniques which can be used as part of the data analysis and on the amount of data which can be used. Often sensor type data acquisition situations have such fast response time requirements that it is necessary to consider off-line analog acquisition and storage techniques rather than direct conversion to digital data formats, see Stacy and Waxman (78).

I. 6 Applicable Statistical Analysis Techniques

It is usually in the areas of questionable, missing, or excessive amounts of data that the analysis techniques interact with other computer implications with greatest difficulty. For the statistical analysis techniques we must take into account the class of application and whether or not we have to be concerned with questionable, missing, or excessive amounts of data. If the application involves prediction most statistical techniques will be satisfactory if the missing data or questionable data are scattered but not present or prevalent in the most recent time periods which may be given the most weight in a prediction technique. If the application involves, say, summarization, most techniques used will be satisfactory if the missing data or questionable data are adequately scattered. In either type of application, if the amount of missing or questionable data is large we lack good statistical techniques. If the questionable or missing data are concentrated, some applications and some techniques can be used by ignoring the data. In some situations the user knows that he has missing or questionable data and can identify the data. The problem then arises as to how to identify such data so that the computer program and associated statistical technique can take advantage of this information. The more difficult situations arise when the user only suspects or is unaware of the questionable or missing data. The problem then arises as to finding appropriate statistical identification and analysis techniques as well as computer programs so as to have useful bypass techniques.

When statistical analysis was performed by hand by one competent in statistical analysis, there was some chance that the questionable or missing data would be detected. To date we have had great difficulty in translating this insight into sound statistical identification techniques and computer programs. When the total amount of data is large, the complexity may also increase in terms of the computer program and the

consequences of the cost and time to analyze the data. These considerations may seriously restrict the kinds of statistical techniques which can be considered.

I. 7 Computing Algorithms

It is possible that perfectly equivalent (in the mathematical sense) analytical specifications of a statistical technique, may give rise to computational algorithms which vary from being useful to worthless. Even in the simple case of computing a variance the representation of $\sum(X_i - \overline{X})^2$ as $\sum X_i^2 - m(\sum \underline{X}_i)^2$ can have serious consequences. The use of the second representation can give rise to overflow in the computation of the sum of squares whereas the use of the first representation usually avoids this possible pitfall, but at the expense of making it necessary to process the data twice in order to have the sample mean available during the second pass of the data. Often the differences between correct mathematics and meaningful computing is lost or ignored and yet constitutes a critical basis for data analysis involving a computer.

The choice between correct and equivalent mathematical representations and good computing is reflected or seen in the selection of the specific algorithm to perform the statistical analysis. Until now, the selection of the algorithm could be based on whether or not it was a precise computational procedure and whether or not it was economical. However, in some applications involving severe response time requirements at an acceptable cost, it is necessary to develop or apply completely different criteria in selecting an algorithm. For example, it may not be technically feasible or economically justifiable to require that all the data be processed twice before arriving at a decision to include or exclude a data observation.

I. 8 Computing Facilities - People, Hardware, and Software

This factor could constitute my entire presentation. Clearly the sophistication of the user of a computer has serious implications for the people who provide computing services in terms of hardware or software. As the sophistication of the analytical techniques increase, it is often necessary to require a disproportionate amount of computing time and

program documentation to ensure that the techniques are correctly used. While some people who strongly expound the virtues of interactive time-sharing assert that a user can be relieved of the need to be concerned with excessive amounts of documentation related to the use of hardware or software, it is my opinion that it is precisely in these areas that the greatest strains have been and will continue to be placed on the application or use of computers as instruments in data analysis. It is with these brief remarks that I would now like to turn to consider briefly the historical developments of statistical computing.

II. SOME HISTORICAL CONSIDERATIONS

Current aspirations involving computers as instruments for data analysis include the demand for highly interactive time-sharing computing with graphical input/output and the demand for one or more flexible and easy to use statistical computing languages, of a possibly conversational nature, see for example, Chambers (15), Claringbold (17), Colin (19), Cooper (22), Dixon (27), Feingold and Frye (31), Gower, Simpson, and Martin (38), Kronmal and Yarnall (53), Miller (57), Schatzoff (74), and Tauchi (80). A review of the historical developments in statistical computing may help us understand current aspirations, and, in turn, these current desires may help us interpret history. It ought to be understood that my historical review does not claim to be complete and is subject to my own errors of observation and interpretation. Volume 16, No. 2 of Applied Statistics has some useful historical information, in particular, Chambers (15). In preparing this section, I have found it necessary to consider several views because I found no single view satisfactory, and they are presented below because I found them helpful to me.

II. 1 Exponential Growth in the Number of Programs

To understand the use of computers in data analysis, it can be helpful to review the documentation of programs which have been developed.

On first glance, one could notice the exponential growth in both the number of computers and the number of statistical programs. However, on closer inspection I found it questionable whether or not there has been exponential growth in the capabilities of statistical programs, even though I believe that the use of computers for statistical computations is still increasing exponentially. During late 1956 and continuing through most of 1957 I tried collecting descriptions of all available programs involving statistical computations This task was undertaken as part of the preparation of my talk, entitled "The Role of Existing Computer Techniques in Statistics and Quality Control" given in December 1957 at the Thirteenth Annual Princeton Conference of the American Society for Quality Control, Princeton University. Although I had collected documentation on several hundred programs, I found the amount of overlap and redundancy in the purpose or scope of the programs amazing, even after taking into account the influence of the number of distinct types of computers. For the published version of the talk, I put together a condensed list of programs which essentially covered the capabilities represented by the entire list of programs which I had collected, after excluding programs with known design flaws or invalid implementations. Somewhat surprisingly, I found it necessary to list titles of only 15 programs, 7 written for one computer and 8 for another. Fred C. Leone, working in conjunction with members of a committee on Mathematical Tables of the Institute of Mathematical Statistics, published in 1961 a 225 page document consisting of "Abstracts of Statistical Computer Routines" (54). In reviewing the document one can easily see that there had been an exponential growth in the number of programs. However, what was true in 1957 was even magnified by the time of the appearance of Leone's publication in 1961; namely, a proliferation in the number of programs without a comparable increase in the capabilities to do new, different, or better types of statistical analysis. Leone summarized his conclusions on page vi by noting the limitations in the programs, their

write-ups, and the quality of the computational techniques. In spite of the subsequent appearance of even more programs, with relatively few exceptions, Leone's remarks are still appropriate. (However, with the appearance of computers with larger memories, more flexible input/output equipment, and easier to use programming languages, some of the unnecessary program limitations have disappeared.) In 1957, see (60), I found it useful to classify the use of computers in statistics into four categories: 1) Administrative Control*- use of computers to create a sequence of instructions for individuals or the computers to aid in the collection of data, 2) Data Processing, 3) Technical Computations, and 4) Simulation Research - use of a computer to generate data according to a specific model so as to create pseudo experiments on the computer through use of the model and to use the data to perform subsequent analysis. Except for category one, much development and computing has taken place since 1957 in activities which can be classified into these categories. However, the failure to see adequate development take place in the first category has meant that some very important aspects of data analysis related to data collection and control have not benefited as much as is possible from improvements of hardware and software. One explanation for this lack of emphasis is the failure to look at the entire process of data analysis; another is the failure to understand the iterative nature of data analysis. This last aspect will be considered again in subsequent sections.

Looking at the development of statistical computing in terms of the number of programs available is of questionable value, especially if there is not some evaluation of the quality of the programs and their contribution to extending the scope of applications. However, I have found this type of evaluation very difficult without relating the programs to the multi-dimensional components cited earlier, Section I. I do believe that an explanation of the redundancy and proliferation of programs (ignoring the "not invented here", NIH, factor) can be given in terms of the

*Would now most likely use the term suggested by J. W. Tukey, "Collection Control.

explanations considered in the next sections. Computer user organizations have done much to avoid proliferation and to help increase the scope of available programs. In this regard, Tauchi in 1965, see (81), has put together a comprehensive 65 page document containing abstracts of statistical programs for use on the IBM 7090/94.

II. 2 Specialization and Generalization

Specialization and generalization of the capabilities needed to enable a computer to be an effective and efficient too in data analysis have been taking place concurrently. Some aspects of specialization with regard to individuals and fields of application can help explain the proliferation of essentially redundant programs. Financial limitations or the specific hardware and software facilities available at a particular time can help explain in retrospect the specialization noted in some programs. Some specialization may have taken place to try to realize economy of computer execution, simplicity in programming and its documentation, or to be sure that a required response time requirement could be met. Another type of specialization, which is often vital to progress, involves detailed microscopic focusing on a particular application. In the need to find a particular solution for a specific application, new techniques are developed but lack of time or adequate understanding precludes generalization at the time.

Some types of generalization permit abstraction and the subsequent reduction of complexity. For example, viewing many seemingly distinct designs of experiments or analyses of variances as special cases of a generalized linear hypothesis can lead to the development of useful insight. While this type of mathematical generalization may be helpful, it does not necessarily promote economical programs or programs which are easy to use or understand for a broad class of users. The iterative and artful aspects of data analysis, as opposed to its analytical aspects,

do not appear to be readily amenable to some types of generalization. With respect to program development, some generalizations have become possible with the improvement in the capability of equipment, increased price/performance of the equipment, and the availability of compiler level programming languages such as FORTRAN and ALGOL.

In the area of statistical computing, the generalization of programs has been inhibited by the lack of effective general purpose algorithms, but much progress has been made in the past few years for analyses involving linear models, see for example Bock (8), Cooper (23), Hartley (40), Nelder (69) and (70), Oliver (72), Schlater and Hemmerle (75), Wilkinson (89), and Yates and Anderson (93). With regard to the programming aspects, the difficulty of generalization has often been traced to factors involving the specification and use of the actual data by the program. We still lack effective and easy to use methods for describing data and its structure. Generality of algebraic results do not necessarily give rise to good computation procedures and it may be necessary to use more specialized procedures; for example, in applications involving missing data, see Healy and Westmacott (43). Often, programs or program packages for which claims have been made about their great generality are in fact of general use only if the data can be organized or separately pre-processed to be input to the program or package.

With the availability of more general purpose programs in the past four years has come the need for improved documentation since the user of the program is, with few exceptions, no longer the programmer of the program. Furthermore, the user of a general program usually needs to specify the selection of options available in the program. The selection of program options can be very time consuming, except when compared to the problems of writing a special program. The use of options can be a source of error but recently this aspect of program design is receiving attention so that the program includes consistency checks related to the selection of options. Many people are optimistic that the availability of general

purpose interactive programs will make it easier to select options of a general program. Some of this optimism will be subject to severe strains in the next few years.

There is another type of generalization that has been taking place in the development of programs; namely, the sharing of utility programs. This is a productive type of generalization. It has been observed that some input, output, or computation activities are identical for a large number of diverse data analysis applications. These capabilities can be provided by the development of highly specialized utility programs which are included only once as part of a general statistical data analysis package, instead of being duplicated in each program of the package. This development aspect will be given additional attention when the design of STATJOB is reviewed in Section V.

The question of what is general and what is specific is often a relative matter. However, in comparing the programs related to data analysis developed in the 1950's and early 1960's and those developed in the past five years, one can find many relatively "general purpose" packages or systems which on closer inspection are seen to be collections of individual programs written in a more general programming language. With very few exceptions one cannot detect a real generalization in terms of extended capabilities for data analysis.

II. 3 Software Development and Its Phases

Some of the generalizations that have taken place in the development of programs and the avoidance of unnecessary redundancy in programs can be directly attributed to transitions that have occurred from the use of machine language coding to the use of assembly language coding (one mnemonic program instruction - one machine instruction) to the use of compliers for higher level languages coding (one program instruction - many machine instructions). To ease the programming burden interpretive programming systems were developed, for example the general interpretive system

developed in the middle 1950's at Bell Telephone Laboratory under the direction of V. M. Wolontis (92). One of the few interpretive systems developed at about this time specifically for statistical analysis was SIS (Case Institute of Technology Statistical Interpretive System). With the appearance of higher level languages, particularly FORTRAN, interest in interpretive systems declined because of their limited flexibility. However, with the development of faster CPU's and central memories, and with the awareness that the percentage of CPU utilization is low, interest has renewed in interpretive systems as part of a general interactive capability in time-sharing systems.

Although, in general, well-written machine-coded programs will execute faster than programs developed by the other means, with few exceptions much less time is required for programming and testing of programs written using compilers. Furthermore, recent compilers, using optimization techniques, are able to provide codes which in most areas of application (excluding Input/Output) and for most programmers are as efficient as programs written using machine coding. The use of compilers as opposed to other programming approaches to write a program has had the advantage of making it somewhat easier to transfer the program to another computer, thus saving effort and avoiding redundant programming.

In considering various ways to summarize how people have proceeded in the historical developments in terms of programming which take into account the components cited in Section I, and in considering how to summarize the design objectives of STATJOB to be presented in Section V, Mr. Peter Wolfe suggested the use of the following four phases as a basis for a historical summary.

1. Initial Phase
   A. Goal: Implementation of particular statistical analysis technique for specific environment, user, data, analysis, application and computer.

B. Solution: Program is coded in any way possible for given problem.

2. Improvement Phase

A. Goal: Improved implementation and reliability of program for a more general environment, variety of users, data, etc.

B. Solution: Program is coded (often in a high-level language) to provide for a variety of data; control specification provides alternate analysis and outputs; more error control included in program.

3. Collection Phase

A. Goal: Standardization and unification of usage of programs for a number of statistical analysis techniques; carry-over between programs.

B. Solution: Collection and standardization of a number of "improved" (Phase 2) programs. Some common components used by more than one program. Limited carry-over of data between programs provided.

4. System Phase

A. Goal: Provide broad analysis and data processing capability for wide variety of users, applications, data, etc.

B. Solution: A system of programs unified and consolidated so that system dominates programs. Higher level programming language used as much as possible. System is controlled through use of a flexible control language. The user can define an analysis involving use of more than one algorithm.

One may question why so much attention has been focused on programming and programming languages. I believe the availability of these

languages helps explain the present state of computing. Along with the improvements that took place in programming languages, one can observe the development of sophisticated operating systems to schedule the control of the computer. Collectively the improvements in programming languages and operating systems have provided increased efficiencies in equipment utilization and some related improvements in general in the flexibility of programs. However, in data analysis applications these developments collectively have inhibited the use of some types of data formats and even more important have moved the user away from his data, the analysis techniques, and an adequate understanding of what is happening to his data while it is being processed. Furthermore, in striving to achieve efficient utilization of equipment, some limitations in operation were introduced which have inhibited the user's ability to perform iterative data analysis in a simple manner, especially if the amount of data is large. Thus while programming has become easier and while efficient use of equipment has improved, we have not seen a corresponding increase in the number of new and improved types of data analysis. Additional possible explanations will be considered in later sections.

To complete the historical discussion in terms of software development, two tables are given below. The first table, Table II. 1 contains a summary of the implications of the mode of coding on the programming effort. The second table, Table II. 2, represents an effort to summarize the historical development in terms of the sophistication that has occurred in the packaging of programs, their data input/output capability, their ease of permitting iterative uses of the data in a single computer run, and their susceptibility to modification.

Table II. 1

A Comparison of Programming Approaches

Mode of Coding	Ratio of Program Code to Machine Instructions	Programming Effort	Transfer-ability
1. Machine	one to one	very great	almost never
2. Assembly Language	one to one	great	rarely
3. Interpretive	one to many	considerable	almost never
4. Compiler	one to many	reasonable	with some effort
5. Interactive Time-Sharing	one to many	reasonable	with effort

Table II. 2

A Comparison of Packaging Approaches

Mode of Packing	Input/Output Handling	Permits Iterative Data Handling	Susceptibility to Modification
1. Special program	depends on language, usually limited flexibility	almost never	great effort
2. Library type subroutine	almost never any capability	never	reasonable effort
3. Application system under standard operating system	limited flexibility	rarely	with difficulty
4. Application system with own operating system	implementation dependent	rarely	great effort
5. Language system under standard operating system	limited flexibility	rarely	great effort
6. Interactive time-sharing	limited	some	great effort

II. 4 Development of Algorithms and Computing Procedures

In reviewing the very early use of digital computers for statistical computations, one can see that most procedures were direct translations of those used with desk calculators or presented in textbooks. Better techniques were considered as soon as there was awareness of questionable results, for example overflow in computing $\sum_i X_i^2$. The need for better techniques would arise for one or more of the following factors

truncation error
discretization error
round-off error
imprecise computational performance (hardware and/or software)
ill-conditioning of the data matrix
missing observations
scaling of observations

With the introduction and availability of floating-point arithmetic instructions on most in the late 1950's computers, people tended to ignore most of the above sources of error in simple statistical computations, and it was only after considerable failures or mis-analyses that people in sufficient number again began to pay attention to these sources of error in statistical computation. Although numerical analysts were doing considerable research in this area, e.g. Wilkinson (90), it is a very recent situation to see special attention paid to statistical computations, even for such "simple" descriptive statistics as means or variances. In this regard, some very interesting considerations have appeared in the past two years. See Gentleman (34), Goldberg (35), Neely (67), Rodden (73), and Van Reeken (87).

The special computational problems related to least squares analysis and the possibility of ill-conditioned design matrices in regression analysis has only begun to receive the attention it needs. In 1958 Box and I

in the design of a non-linear estimation program implemented by Booth and Peterson (see Booth, Box, Muller, and Peterson (10)), introduced the use of the eigenvalues as a tool to see the extent of ill-conditioning in the computation. Taussky in (82) drew attention to the fact that $X'X$ had worse conditioning than X, yet most textbooks still recommend the use of $X'X$ as a computational approach without any cautions. One now can find an impressive literature appearing which is specifically addressed to the problems of regression and least squares analyses, see Bauer (5), Björck (7), Golub (36), Golub and Wilkinson (37), and Jordon (52). The use of iterative improvement techniques is a significant contribution, see Forsythe and Moler (32). For reasons such as mathematical insight and relative ease of computation, least squares analysis has usually been used instead of the minimization of, say, the sum of the absolute deviations. However, the current capabilities of computers now make it possible to consider the more complex methods, see for example Barrodale (4). Beaton (6) did some especially important work for linear models with regard to considering methodology and programming implications. For the analysis of variance, Bock (9) gives attention to needed multivariate algorithms. (See also references cited in Section II. 2.) The use of transformations to aid analysis has had some significant attention, for example, Box and Cox (12). With the availability of additional theory on residual analysis, see for example Anscombe and Tukey (2), the computer has much to offer in the analysis of data in terms of residuals.

The need for rational approximations to obtain observations or probability values from specified statistical distributions has received considerable attention, in particular Hastings (41), and NBS (66). Techniques to select samples of data were developed by Muller (59), Fan, Muller and Rezucha (29), and Jones (51). The generation of pseudo random observations has received extensive consideration. Two recent historical summaries can be found in Jansson (50) and Muller (64).

It has been recognized by many people doing data analysis that stepwise regression procedures can be both very useful and fraught with computational dangers. Recently, this topic has begun to get the attention it deserves, see for example the review in Anscombe (1), and the paper by Lindley (56). It is interesting that this is an area where the computer has enhanced data analysis even though the analytical techniques employed are not necessarily based on solid mathematical justifications. A justification for this type of approach has been lucidly presented in Tukey and Wilk (85). Starting with the work of Box and Coutie (11) in 1956, significant attention and progress has been made in developing routines for non-linear estimation, see for example Non-linear Least Squares (GAUSHAUS) by Wolfe (91).

Another type of significant improvement in computational efficiency has come about because of the reformulation of analytical techniques in terms of efficient computing algorithms. A particular example is the development of the "Fast Fourier" techniques stimulated by the Cooley-Tukey paper (20). In this application, as with others, the amount of computer memory needed may be important in influencing new techniques, see Singleton (77). Programs to handle time dependent events are beginning to get needed attention, for example, Lewis (55).

The old cliche, "A picture is worth a thousand words" has real significance in data analysis. The plotting of residuals can provide exceedingly useful insight. The use of probability plotting can be of considerable value too. These two types of pictorial representation are rather straightforward computational and programming applications which are now in extensive use. However, the latter has depended upon the development of analytical tools such as done by C. Daniel (25) and Wilk and Gnandesikan (88). Much interest has been expressed in the use of contour plotting, for example in the display of likelihood contours. Depending upon the device being used to display the contours (printer, plotter, graphics device), there is analytical complexity as well as programming complexity.

See for example IBM (47). For reasons of accuracy and costs, much more work needs to be done on the analytical side of the pictorial representation of data.

The point of this section has been to emphasize that some significant progress has been made in the past few years in bridging the gap between good mathematics, statistics, and good computing for statistical data analysis. The progress is encouraging. Depending upon what one considers to be data analysis, one can demand more statistical theory, more statistical techniques, and more analytical improvements in algorithms, but it is suggested that one read Tukey (83) and Tukey and Wilk (85) to see why there may be limitations to a purely mathematical approach to data analysis.

II. 5 Some Concluding Remarks Leading to the Present

An attempt has been made to outline in summary form what has happened in the development of computers as instruments in data analysis. Having looked at the number of programs, programming, and analytical techniques, it is hoped that these views will collectively provide some useful insight into where we are in the use of computers in data analysis and where we can hope to go. The digital computer very early in its history became an effective and efficient replacement for the desk calculator for many types of extensive statistical computations. Subsequently, much significant progress was made in the development of hardware and software, but a comparable growth in the quality and capability of programs for data analysis has not been observed. As mentioned earlier this lack of comparable progress may in fact be due in part to the non-analytical aspects of data analysis, see Tukey and Wilk (85). Another explanation is that development emphasis has been placed on efficient use of computer resources and the efficient sharing of programming effort. Recently some attention has been given to determining the adequacies of the underlying algorithms needed in statistical analysis.

In the very early days of digital computer use the programmer was also the one who most likely had the data analysis problem and was also the one who most likely ran the problem on the computer. Thus attention was not focused on the importance of having the programs permit iterative type data analysis because equipment limitations forced a certain type of iterative or multi-step processing which required the user to handle the data more than once. The subsequent amazing improvements in hardware and software made it possible to eliminate some of the undesired steps in the processing. However, in the process of achieving such improvements the need to protect or expand the capability to perform iterative types of data analysis was not adequately considered. Iterative-type data analyses should be much simpler and effective with the availability of interactive time-sharing type computing. Recent developments in this area have been essentially excluded in this historical section because these activities are current and will be considered in the sections on STATJOB and the future. They have also been excluded because I lack sufficient experience to look at them with any type of historical perspective. Finally, the desire to avoid a premature evaluation was prompted by the following quote from Tukey and Wilk (85), page 695, "As in any other science, what is done in data analysis is very much a product of each day's technology. Every technological development of major relevance ... has been accompanied by a tendency to rediscover the importance and to reformulate the nature of data analysis."

III. SOME AVAILABLE LARGE SCALE STATISTICAL PACKAGES

To provide some feeling for the number of available statistical program packages, Table III. 1 below contains a list of some of the major packages which operate under or as a part of a batch processing operating system. Only those packages for which there has been generally available program documentation or descriptions in formal publications have been included in the Table. Many of these packages have been referred to in Volume 16, No. 2, of Applied Statistics, 1967.

Table III. 1

Large Scale Statistical Packages Operating in Batch Processing Environment

Identification	References
ASCOP	Cooper (22)
BMD	Dixon (26)
BOMM	Bullard, et. al. (14)
COOLEY/LOHNES	Cooley and Lohnes (21)
COSMOS	O'Brien (71)
GENSTAT	Nelder (68)
ISSS	Fan and Scheidemantle (30)
MEDCOMP	Sterling and Pollack (79)
METO	Craddock and Freeman (24)
MSP	Clyde, et. al. (18)
OMNITAB	Hilsenrath, et. al. (44)
P-STAT	Buhler (13)
SSP	IBM (49)
SSUPAC	Dodson, et. al. (28)
STATPAC (Goodard)	Shannon and Henschke (76)
STATPAC (UNIVAC)	UNIVAC (86)
STATJOB	Hutchins, et. al. (45)
STORM	IBM (48)
SUMX	Champomier (16)
TSAR	Gabor and Carlitz (33)

IV. IMPLICATIONS OF LARGE DATA FILES

General interest in computer oriented data files (now often referred to as data bases) has existed since reliable magnetic tapes and tape drives have been available as standard equipment. Additional significant interest in data bases has been generated with the general availability of

pseudo random access devices such as drums and disks. Computer based data files is a component of computer data analysis that has serious ramifications. It is my belief that statistical techniques are needed to provide compact and easy to use data files which are useful summarizations or descriptions of more extensive data bases which have been analyzed in a background mode so as to keep the cost of analysis within reason. For these reasons and because the consideration of large computer based data files provides illustrations of the inter-relationship of the components of data analysis cited in Section I, a separate section is being devoted to this topic.

Within the past few years many areas of application have gone from a "data poor" to a "data rich" status. One reason is the extensive technological improvements which have facilitated the acquisition of data either without a computer or with a computer. Another reason is the improvement in the reliability and price/performance of auxiliary or secondary data storage devices for computers. The old cry of needing more data to perform a meaningful data analysis now sounds rather hollow for some applications. This rapid change in status has created some interesting challenges and has made it necessary to reformulate what constitutes meaningful data analysis.

Some analytical or processing techniques which were developed or used to handle small amounts of data provide some rather disappointing, questionable, and very costly ways to handle large amounts of data. In addition to emphasizing the need for additional statistical techniques to cope with the data rich situations, the purpose of this section is to explore some of the implications of using computers to cope with large amounts of data as part of a data analysis.

IV. 1 Consequences of Handling Large Data Files

As the size of a data file increases, it is usually safe to assume there has been a corresponding increase, or investment, in the file in

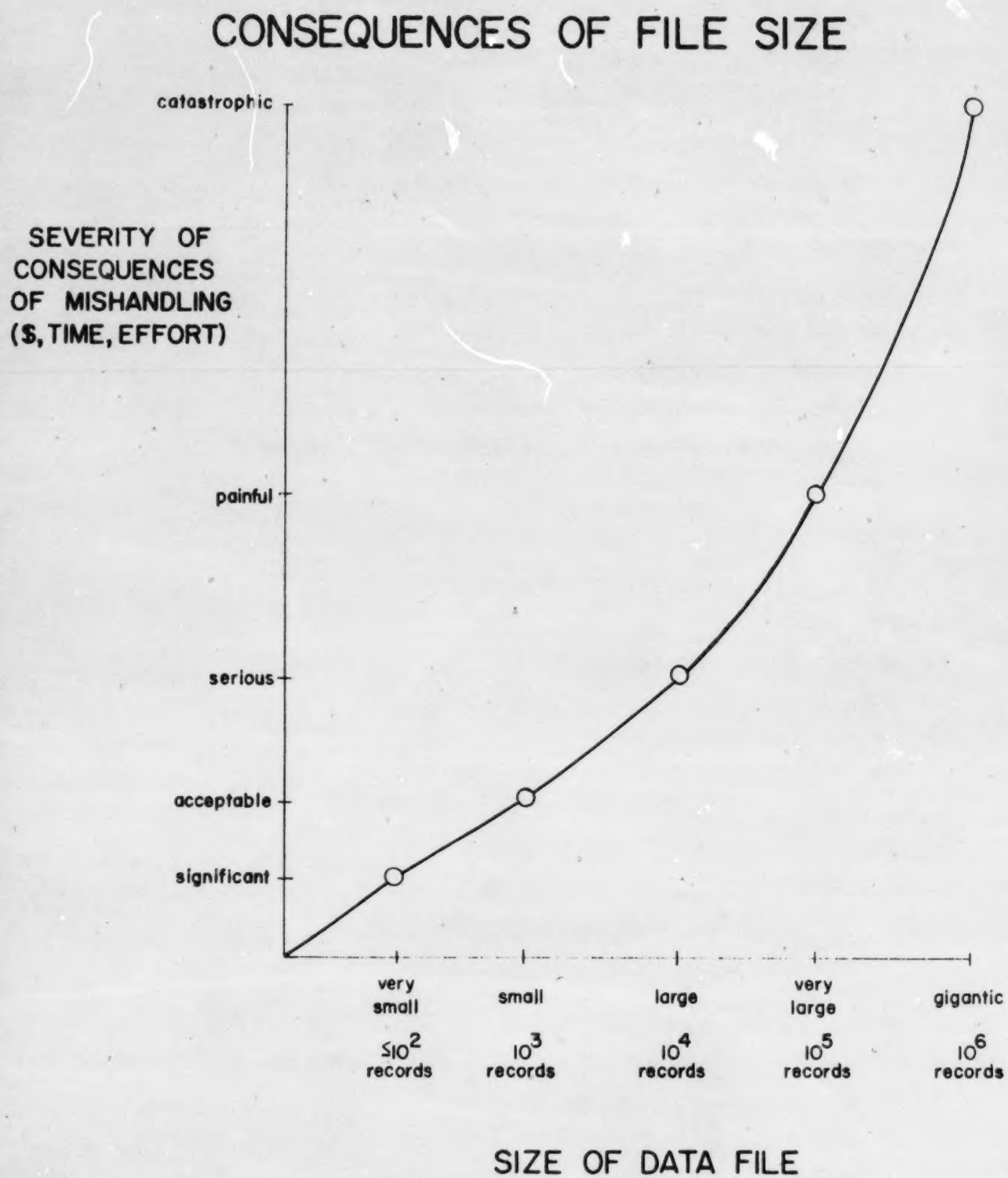
terms of time to collect the data, effort to collect the data, and cost of collection and storage. A poor technique for the identification, storage, analysis or retrieval of a data file has increased consequences as the size of the file increases. A poorly posed analysis or poorly posed retrieval request takes on increased significance as the size of the file increases in terms of cost, time, and effort. As an example of the severity of consequences of mishandling a file, as a function of its size, think of the cost, time and effort to search for a non-existent item in a file. Thus, in my opinion, whether or not one is focusing on the collection, storage, maintenance, retrieval, access to or analysis of a file, the consequences of erroneously handling the file increases exponentially as the size of the file increases. These thoughts are summarized in Figure IV. 1. In the following subsection attention will be given to the mode of data acquisition.

IV. 2 Serial, Parallel, and Hybrid Modes of Data

It is useful and necessary to take into account how the data will be made available for analysis purposes. We will let x_{ij} denote the ith observation on variable, or factor, j. The data will be said to have been collected or available in a serial mode if for each j, when $j = 1, 2, \ldots, J$, successively, the observations x_{ij}, $i=1, 2, \ldots, I$, are available according to order i, (variables by occasions). The data will be said to have been collected or available in a parallel mode if for each i, where $i = 1, 2, \ldots, I$, successively, an observation is available on each of the factors, x_{ij}, $j=1, 2, \ldots, J$, according to order j, (occasions by variables). The data will be said to have been collected or available in a hybrid mode if it is neither in a serial or parallel mode of identification by occasions. The hybrid mode is often present in data collected for example in a n-way analysis of variance, with $n \geq 3$.

For small amounts of data one can usually afford to accept and process the data regardless of its mode, assuming each observation can be correctly identified. In the past, much of the identification of data was

Figure IV. 1


CONSEQUENCES OF FILE SIZE
catastrophic
painful
serious
acceptable
significant
SEVERITY OF CONSEQUENCES OF MISHANDLING ($, TIME, EFFORT)
very small
small
large
very large
gigantic
≤10^2 records
10^3 records
10^4 records
10^5 records
10^6 records
SIZE OF DATA FILE

derived from or could be assumed from its order of presentation. This mode of data collection and presentation is simple but provides no basis for checking on whether or not the data were correctly presented according to order. This approach is a direct carryover from using data worksheets with desk calculators. Most current statistical analysis programs are, unfortunately, based upon this approach. It does simplify the program and the preparation of data, but unfortunately does not make it possible to make full use of the computer for checking or controlling the data.

For large amounts of data the mode of the data can have serious consequences with regard to its use in an analysis. For example, there may be good practical reasons why the data are presented in serial mode, yet for efficient computation on the data, they could be used more effectively in a parallel mode, one observation at a time on each of the variables. Thus additional processing and intermediate storage may be required unless, for example, we could work with X' instead of X. As the size of the file increases the importance of its order and the quality of the data identification becomes exceedingly important. In the hybrid mode case, it is absolutely necessary to have each item identified not only to avoid errors but also to permit meaningful computations.

As the file size reaches some critical amount relative to the amount of main memory available in the computer for work area and the amount of fast access auxiliary memory available to hold intermediate results, the techniques for data analysis and data identification may have to change significantly. Some additional remarks will be made on this point in the remaining parts of this Section.

IV. 3 Serial, Random, and Pseudo Random Access to Data

The method for storage and media upon which data are stored in a computer have serious consequences in data analysis both with regard to how much identification information needs to be included with the data and whether or not one can get random access (time required to access each

item is identical) to the data. When data are stored on magnetic tape (or punched cards) they have to be processed and identified in a serial order. Thus the access time to a particular item is a function of its position. However, the advantages of a serial storage device such as a magnetic tape are that it preserves certain positional information that does not have to be included in the data and that if the data can be used in the order they are stored on the tape, then considerable computer-system efficiencies can be realized. Good magnetic tape drives also make it possible to provide certain write and read checks with the data are processed.

There are some applications where the data must be made available or processed in a random order, e.g. random access to a file to meet an on demand response time requirement involving finding a record in the file or in situations where only a small amount of data can be held in main memory, or it must be brought in from auxiliary memory in a random order. If the amount of data is large relative to main memory ("main" memory can; by definition, be accessed randomly), then it is necessary to store the data on devices with relatively fast access times and which permit essentially random access, for example drums or disks. When data are stored on drums or disks one has essentially random access, but there are some variations in the time required to access an item, this will be identified as pseudo random access. With random or pseudo random access, it is necessary to have adequate identification of each item of the data file. As mentioned earlier this adds complexity to both the program and the collection and storage of the data. However, for those applications with large amounts of data and with a random or pseudo random access requirement, one usually is forced to consider the required additional complexity. Most pseudo random access devices do not provide hardware which permits write and read checking in the sense of a tape drive. If one needs this kind of

reliability, this adds additional program complexity. The cost of random or pseudo random access storage, to date, is so expensive that few data analysis problems can afford to be approached from this point of view. Additional research is needed to assist in the development of techniques to aid in the effective storage and retrieval of items stored on such devices.

IV. 4 Backup and Error Recovery

The subject of how to recover from a computer malfunction is a separate subject. However, when one is doing a data analysis involving a large amount of data, one cannot usually afford to recreate the data. Therefore, as part of good data analysis practices, there should always exist a backup copy or a set of files which collectively could provide economical reconstruction of the data file. Error recovery procedures are also needed so that the entire data analysis does not have to be done over in the event of an error. In this area, data stored on magnetic tape makes both backup and error recovery much simpler than when random or pseudo random access storage devices are used. To date, almost no general purpose statistical programs or packages provide backup or error recovery capabilities. Very few computer installations provide this capability through the operating system. As data analysis on large files becomes more common, the need for work in this area will hopefully receive the attention it deserves and needs.

IV. 5 Passing Program Over Data

In data analysis applications involving large files, one cannot usually afford the cost of making repeated passes through the file to achieve the analysis, yet in many applications and for many techniques this appears to be a necessary requirement. However, if the data are adequately identified, many applications can be formulated so that several different data analyses are performed while processing through the file.

Several recent statistical packages allow this, for example STATJOB. However, the next extension is to invert our customary way of considering doing data analysis. Thus instead of processing data through programs, think of passing programs over the data in the sense that as each data item is identified, it is analyzed relative to all needs at that time by sequentially obtaining and using the appropriate programs. This is important for large files. J. W. Tukey is the one who expressed this hope to me many years ago. With adequate data identification and with programs to perform the identification and selection, it is possible to achieve this objective. By the development of sequential identifical and selection techniques, see Fan, Muller, and Rezucha (29), and the use of a "link order logic" developed by Fan and Muller, see Muller (61) and Fan and Scheidemantle (30), it is possible to exhibit this capability of passing program over data. Except for this work, this area still remains essentially ignored.

IV. 6 Sampling of Files

Statisticians feel comfortable about recommending the use of samples to acquire data. Yet few recognize the need to use data stored in computer usable form on a sampling basis. This need will continue to grow as data files increase in size. Somehow there is a feeling the computer is fast enough to use all of the data.

Using the ideas of passing program over data and some sequential selection techniques developed by Fan, Muller, and Rezucha (29), and Muller (63), it is possible to select samples of any known kind with a program using these techniques, namely ISSS (Information Selection and Sampling System) by Fan and Scheidemantle (30). When the size of the file is large, the need to avoid processing the entire file to determine the size of a stratrum has real value and can be achieved by the techniques cited above. It is my hope and belief that data analysis on large amounts

of data will be approached from a sampling point of view. To date, we tend to insist that "all" the data be included in a computer data analysis if "all" the data are available. I hope this is not a permanent situation.

V. STATJOB

STATJOB is a particular system for doing statistical data analysis on a digital computer. It is a system in the sense that it handles all of the computation activities needed to: 1) input controls and data, 2) select from among the analysis routines available and perform the analysis, and 3) output information and the results of the analysis. A systems approach to statistical data analysis has been taken to obtain efficiency, effectiveness, and generality in using a digital computer. A systems approach has been taken to obtain an integrated collection of functional services, to realize economies from the consolidation of functions, and to learn of the interactions of the functional components.

V. 1 Global Systems View of Data Analysis

The purpose of taking a global systems view of data analysis is to obtain a better and more comprehensive understanding of the factors of data analysis (cited in earlier sections, particularly Section I) than would be possible by not looking at these factors collectively. This systems view should help to identify the factor interactions and to establish the requirements of computer programs for data analysis to meet the needs of the users. By taking a global systems approach to data analysis we could hopefully better identify what are its inputs, the actions taken on the inputs, and the required outputs for end use or feedback. We would hopefully be in a better position to obtain efficiency and simplicity in data analysis by using a systems approach in allocating resources in design and implementation and in resolving incompatabilities between requirements and goals.

The general components or factors that have to be kept in mind in developing a computer system for data analysis have been outlined in Section I, for example those due to the user, the application, the environment, the data, the equipment, etc. The anticipated or actual goals of the user become the initial requirements of the designer who in turn establishes a set of goals which become the initial requirements of those implementing the design. In a well conceived system the needs for maintaining the system should be considered as part of the design and implementation of the system. For this reason, we consider that the implementation requirements become the initial goals of those maintaining the system. In tables V. 1 through V. 4 are presented summary outlines of the goals of the user, the designer, the implementation, and the maintainer of the system. It should be understood that these four sets of goals interact and create the need for compromises. Figure V. 1 is presented to summarize the interactions and feedback aspects among goals, requirements, operating environment, the user, and the statistical data analysis system.

V. 2 System Design Goals of STATJOB

The goals established for STATJOB include most of those cited above in Section V. 1; however, there are some exceptions. In designing STATJOB we had several major goals not cited above. We desired: 1) to provide a vehicle for testing our design ideas, 2) to have a production version of the system to meet user needs and to evaluate implementation accomplishments, and 3) a test version to permit experimentation. Even though STATJOB has been in production use since late 1966, in some sense the entire STATJOB system can be considered an experimental undertaking to see how far we could go in standardizing selection and use of:

options
input
analysis
output

Table V. 1

User Goals for a Statistical Data Analysis System

Ease of learning system

Ease and simplicity of use

Directness of use

Wide scope of application

Assistance in data gathering

Assistance in editing data
(checking and transforming data)

Flexibility in handling data input

Flexibility in storing data

Flexibility and generality in obtaining data output

Iterative handling of data

Interactive handling of data

Economy and ease of use in terms of multiple passes on same data by different analyses or in terms of use of same analysis on multiple data sets

Economy in general

Efficiency

Reliability

Accuracy

Expandability

Table V. 2

Designer Goals for a Statistical Data Analysis System

Two levels of use
- learner
- expert

Meet analysis requirements

Flexibility in selection of options

Consistency checking of options

Varied data formats for input and output

Easy specification of data formats for input and output

Easy to understand descriptions as part of output

Iterative processing, output of one analysis as input to another analysis, on a selective basis

Iterative processing - same data used by several analyses
- same analysis on several data sets

Adequate diagnostics to user - perform as many diagnostics as possible, even when errors are present

Adequate diagnostics control to system

Standardize when possible use of options for input, analyses, output

Standardize when possible input, analysis, output, and documentation

Modularity in design

Determine and share common utility processes

Optimize program execution

Backup files

Error recovery and restart

Adequate description of performance and checkout requirements

Expandability

Transferability

Table V. 3

Implementors Goals for a Statistical Data Analysis System

Satisfy design objectives

Have multiple modes of selection and consistency checking of options

Determine modularity for input, option selection, processing, output

Isolate unique features of each type of analysis

Isolate utility processes

Layout system to aid

- testing of components
- testing of integration of system
- documentation
- modifications to system

Provide test mode

Provide meaningful test cases

Generation of information on systems use and performance

Provide backup capability, and error recovery and restart

Generation of documentation for

- users
- maintainers

Table V. 4

Maintainer Goals for a Statistical Data Analysis System

Maintain current level of implementation

Mechanism for an audit trail of system changes

Mechanism for comparing new and old version of system

Capability for testing system

Capability for timing

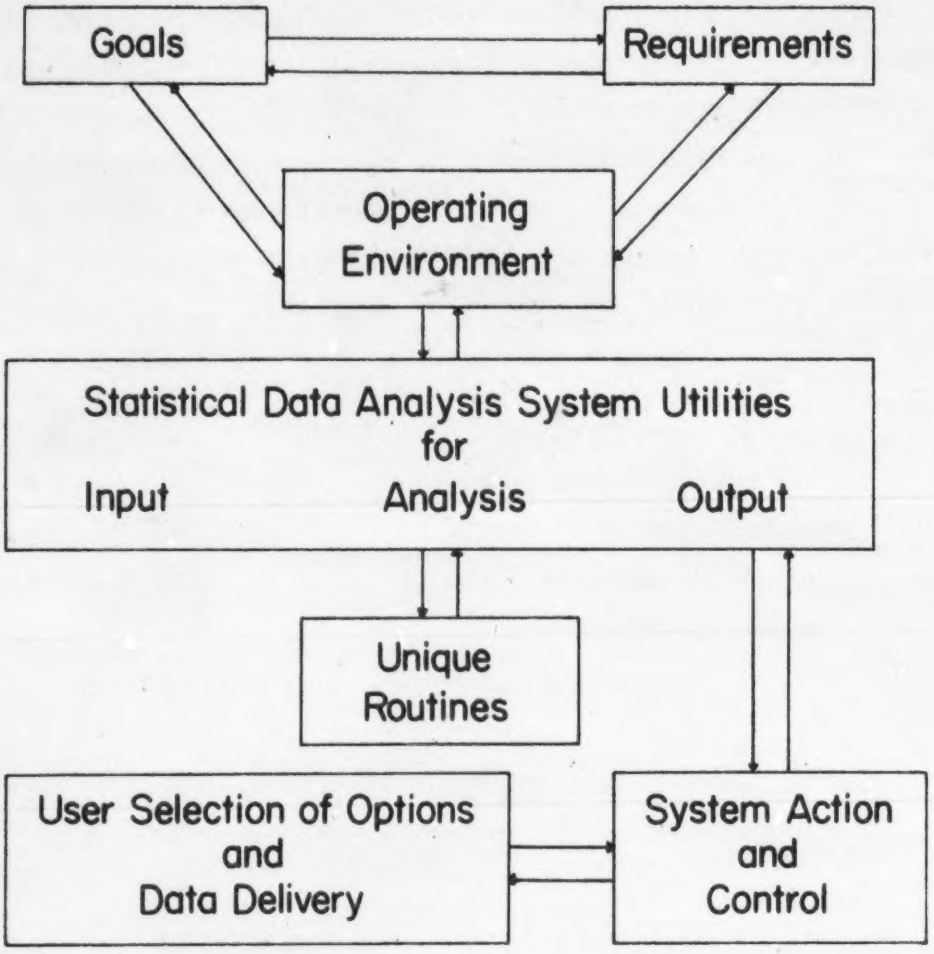
- utility processes
- computing algorithms
- applications

Documentation generation for

- new capability
- changes in capability
- corrections
- improvements in capability

Figure V.1

A GLOBAL SYSTEMS VIEW OF A STATISTICAL DATA ANALYSIS SYSTEM



We also wanted to see how far we could go and how effective we could be in designing and developing a modular system with routines to be used on a utility basis. We also wanted to design a control capability for the selection and consistency checking of systems options which would work in a batch operating mode but would be readily converted to an interactive time sharing environment. We desired to provide data carryover capabilities so that the output of one analysis could be used as the input of another analysis within the limitation of not requiring major changes to the batch operating system. A goal was to provide a system which could be used by either a novice or an expert in the use of computers, assuming both had the required statistical competence. Another goal was to try to make as extensive use of a higher level language (FORTRAN) as possible, to enhance the transferability of STATJOB. Finally, we wanted to be able to accomplish as much data analysis as possible in a single use of any data file.

We were faced with certain constraints, namely:

- operate in a batch mode on the equipment represented by the CDC 1604
- operate as standard jobs under the UWCC operating system on the CDC 1604
- provide statistical analysis capabilities to replace existing routines
- limit machine language programming to memory management and interface of STATJOB with operating system

V.3 Implementation Accomplishments in STATJOB

One of the original design goals for STATJOB was to create a system which would permit evolutionary improvements and changes; for most aspects of STATJOB this goal has been realized at a reasonable price in terms of effort and time. Although the implementation of

STATJOB is evolutionary in nature, it is possible to evaluate its current implementation status. In some areas of implementation, the success achieved has been extremely gratifying. It has been possible to achieve a modular system which makes extensive use of a set of shared utility processes while still being able to achieve a high degree of standardization without restricting the flexibility of the system.

It has been possible to achieve a very powerful and easy to use data transformation language with self diagnostic capabilities which can be used to edit, recode, exclude, or transform the input data to provide new variables according to arithmetical, functional, or logical operations. Although it is not obvious to the user when the transformation language is used, it is compiled into very efficient machine code as required. The compilation is part of the processing of the control information which selects options of the STATJOB system. The user is provided extensive diagnostic information at the level of the transformation language he used, thus assisting him in finding errors, and assuring him that he has not attempted to do illegal operations. The transformation language makes extensive use of the techniques developed for DITRAN, a diagnostic FORTRAN compiler, developed at the University of Wisconsin by Moulton and Muller (58). The availability of this transformation capability makes it possible to perform many types of data manipulation operations while processing the data file as part of the analysis phase, thus realizing savings in computing time as well as flexibility. A more detailed description of this capability can be found in UWCC Users Manual Volume VI and Hutchins, Muller, and Wolfe (46).

The modularity of the design of the system has had significant implications for the extent of the standardization that could be achieved in the documentation of the system for the user. The user documentation associated with STATJOB represents a major investment in effort. Some of this effort was devoted to determining how to standardize and organize the

documentation to aid the user. When compared with other statistical packages, the documentation of STATJOB is much more extensive; however, we believe this detailed documentation is necessary if the user is going to be able to use the system based upon reading its documentation.

The remaining discussion on implementation will be summarized in terms of the major functional components of STATJOB; a much more detailed presentation can be found in Muller, Wetterstrand, and Wolfe (65). Figure V. 3. 1 below contains a pictorial representation of the major functional components.

V. 3. 1 Control Processing

The selection and control of input, analysis, and output options are handled by a set of utility programs identified as the control card processor. At this time, all selection and control are achieved by the use of standardized control cards which in almost all cases but the specification of the data records are used in free field format except for the fields assigned for card identification and control. The control card processor contains routines to check that the control information has been specified consistently relative to the specified analysis required and notifies the user of any errors or potential errors. The control card processor makes it possible for the user to specify models, say for regression analysis in terms of names of variables or equations, or desired tabulations in terms of headings and subheadings. The design objective of analyzing as much of the control information as possible, even when errors are present, prior to terminating a job has been achieved. It has been possible to have the control processor control the use of a specified analysis routine so that an analysis is performed on multiple sets of data or on a single set of data according to several options. These capabilities have contributed significantly to the goal of doing as much analysis as possible in a single use of a data file. Furthermore, to maximize the amount of main memory available for analysis purpose, the

control processing includes the loading of only the required parts of STATJOB.

V.3.2 Input Processing

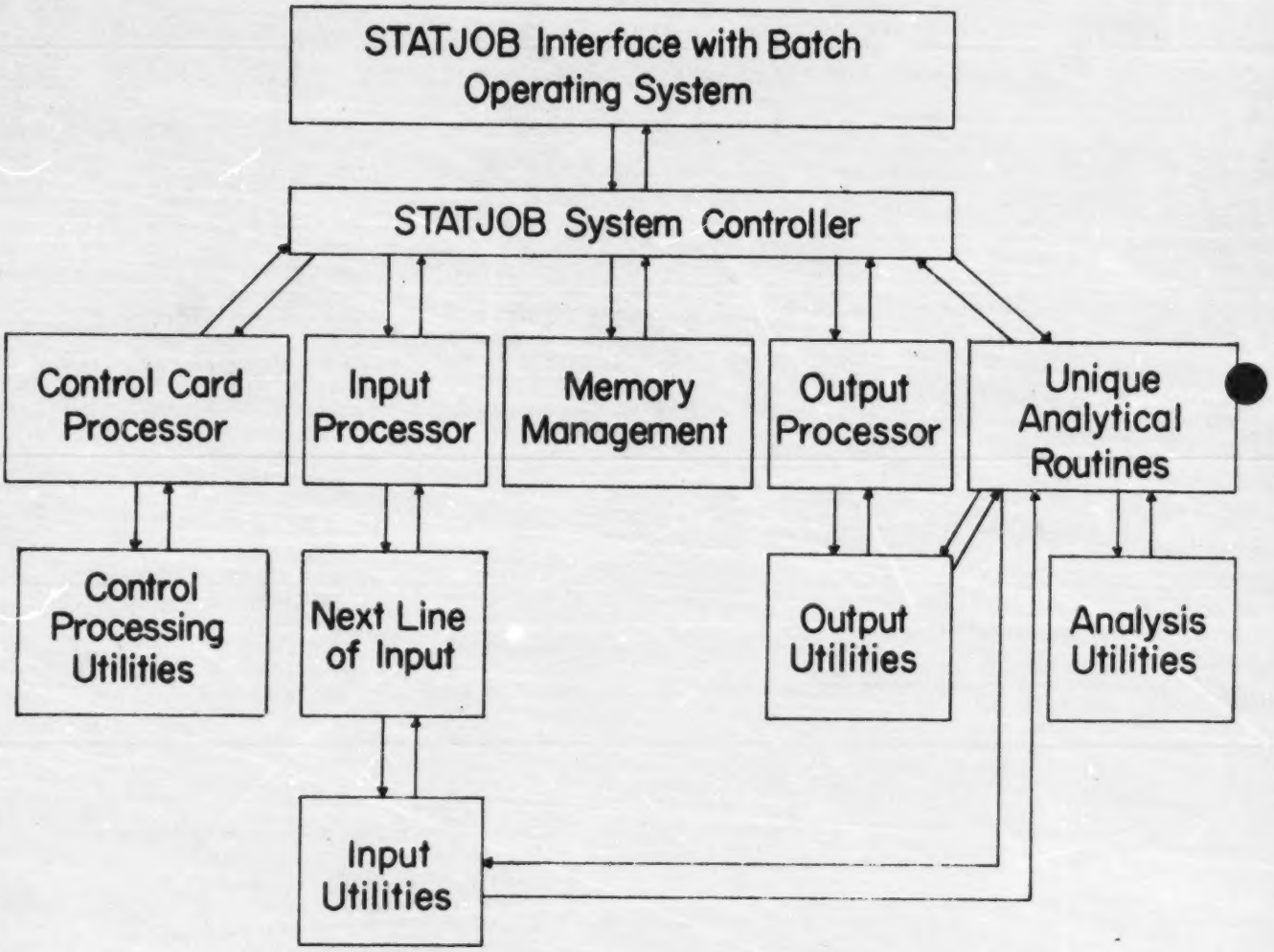
By the use of control options which are processed as part of the input, the user can specify the format of his data covering a variety of data arrangements where the input may be on cards and/or magnetic tape with or without internal labels. The user can select and specify that the input data are to be extensively checked for errors according to say, a sequence order or other set of criteria. He can also specify that some desired editing, recoding, or transformations be performed on the input data. Under some restrictive conditions, the output of a prior analysis can be used as input to the current analysis. The system in general and the analysis routines in particular work most effectively when the data are in what was described in Section II as the parallel mode. Depending upon the particular analysis desired, the system can handle, with restrictions, data in a serial or hybrid mode.

V.3.3 Analysis Processing

The design objective of being able to develop modular analysis routines which could be packaged to make efficient use of utility routines for input, specification of analysis, assistance in analysis, and output has been achieved. Some of the utility routines which were developed to assist or be part of the analysis provide such capabilities as matrix handling, table generation, generation of the statistical level of significance according to the required distribution, for example, the F-distribution. As new analytical capabilities have been added to STATJOB, new insight has been obtained on how to specify the analysis, e.g. model information, how to package the analysis requirement, and how to interface the analysis routines with the other functions of the system. In some cases, as new capabilities were added, the control card processor had to be modified or expanded, in other cases it was the control system, input, processor,

Figure V. 3. 1

FUNCTIONAL COMPONENTS OF STATJOB



or output processor that had to be modified. On the statistical analysis side some initial success has been realized in developing routines that are able to detect and indicate to the user that the assumed required mathematical or statistical conditions are in question; for example, data on fewer variables than specified in the model, ill-conditioning of the data matrix, or correlations in the residuals.

V. 3. 4 Output Processing

As in the case of input processing, control options can be used to select and control output; considerable flexibility exists here. In spite of the various types of analyses which a user of STATJOB can request, considerable standardization has been achieved in formatting the output without denying flexibility to the user. While many utility routines have been developed here, we have not gone as far as we feel we can. As part of the output associated with each analysis, the user receives a readable summary of: 1) all the control information he supplied, including output titles, 2) acknowledgment of the validity of the control information or diagnostic information indicating the errors found in the control information, 3) a data-input summary showing the first and last record processed for each data set to assure the user that he processed the desired data set or sets, and 4) if specified, a systematic snapshot of every kth data record. If transformation or recoding of variables had been employed, or if variables had been named to aid the specification of the analysis or the description of the output, this information is included as part of the output summary. The user can specify printed, punched, or magnetic tape output with or without a label. This output, for some types of analyses, can be in a condensed form for carryover as input to a subsequent analysis, e.g. sums and sums of squares with adequate identification, on cards or magnetic tape.

V.4 Summary of Actual Experience

STATJOB became operational in 1966. The extent of its use and acceptance can be measured in a sense by realizing that 10,414 jobs were processed in fiscal 1967-68 (July 1967 - June 1968). Acceptance of STATJOB has gone well. From the point of view of the users, we have met some of their needs very well. Except for the specification of record formats, the learning of the system has presented few difficulties. The transform language capability has been very successful, although as we expected, the initial (now improved) recode capability was time consuming for the user with jobs involving the recoding of many values of a variable. Although STATJOB is a systems approach rather than a collection of routines, its price/performance for the user has in general been equal to or better than that available from less flexible and less reliable routines.

Our experience with the documentation reinforced our early conviction that extensive documentation is needed and desired by the user. However, we have also learned that a less detailed level of documentation is also needed to assist a user in deciding on the use of a capability or in reviewing the use of the system once one knows how to use it.

From the point of view of the developers of the system, we have learned of the need for performing adequate "field-testing" of a program and its documentation prior to general release. We have learned that it is helpful if in addition to the production version of STATJOB the development version of the STATJOB system can be hidden from the users but available as part of the standard operating system so that changes and new capabilities can be tested as standard runs.

The analytical capabilities currently in STATJOB are summarized by category in Table V.3.1, page 45.

Table V. 3. 1

Summary of Current Analytical Status
of STATJOB

Available

Cross tabulation tables and analysis

Descriptive Statistics

Factor Analysis

One and two way analysis of variance

Regression Analysis

Stepwise regression analysis

Separate data transformation capabilities

In development in order of expected release

Pictorial statistics

N-way analysis of variance*

General orthogonal polynomial fitting*

General orthogonal function fitting*

Sample selection

Non-linear estimation*

Time series analysis*

Limited, multi-step, and full information estimation

Balanced designs

Incomplete designs

Adaptive forecasting

Discriminant Analysis

*Individual UWCC programs now in existence

V.5 Weaknesses of STATJOB

To make the initial system available in 1966, we placed several limitations on the system. In particular, from the user's point of view we restricted the data input to be records with data formats satisfying the conventions in FORTRAN 63 (CDC FORTRAN). Furthermore, we required the use of FORTRAN format conventions for specifying the formats. These limitations still exist but plans have been considered to remove these restrictions which are an inconvenience or cause of difficulty for users. The system does not have adequate capability for handling variable length or arbitrary blocked records. As can be seen by studying Table V.3.1, STATJOB has limitations in the kinds of available analysis capabilities. For example, in the general analysis area we have made available a very flexible and powerful one-way and two-way analysis of variance capability but have not included a general n-way capability. Such a program exists at UWCC but not as part of STATJOB. The computational aspects present no difficulty. However, difficulties do arise when you want easy and reliable ways to allow the user to specify the model or to consider arbitrary or hybrid modes of presenting the data to the program for a specific model. The system lacks a general discriminant analysis capability. Although the UWCC has extensive plotting capability, it is not available directly through STATJOB at this time. As mentioned earlier, the data carryover package has limitations.

From the point of view of the design and internal operation of STATJOB, there are limitations which are a result of compromises to make the system operational in 1966. Other limitations are now apparent based on the experience of implementing or using STATJOB. For example, although STATJOB contains many output utility routines, it was expedient to compromise the development of some utility routines by implementing special purpose output capabilities to meet the needs of a specific type of analysis. At this time, we can obtain elaborate statistics

on the various routines used within STATJOB. However, we do not obtain adequate data which indicates when STATJOB is being used incorrectly or when specified analytical or control conditions are not satisfied. The challenge is to determine how to generate this type of data in an efficient manner. We feel these kinds of use data are vital to determining how to improve the effectiveness of STATJOB, in a meaningful and useful way.

V.6 Future Plans for STATJOB

To overcome the weaknesses of STATJOB, there are some obvious directions for improvement of STATJOB in terms of use, capabilities, and design. These include:

- Additional analytical capabilities
- Generalized and simplified methods for specifying data record formats
- Efficient handling of variable length records or data blocks
- Extensive pictorial presentations
- Link order control to permit general data identification
- Capabilities for efficient storage and retrieval of intermediate results and data
- Chaining of analysis specifications to permit additional iterative types of computing
- Usage data on mis-use of capabilities - analysis diagnostics
- Automatic error restart and recovery
- Interactive use of STATJOB

Although STATJOB was designed for the CDC 1604, it has been converted to operate on the CDC 3600 under the UWCC operating system. It is also being converted to operate on the UNIVAC 1108 under EXEC VIII. Based upon these conversion efforts and the design of STATJOB, it appears that there are fundamentally only three areas that are dependent upon the computer operating system under which future versions of STATJOB might operate. These are the system interface, memory

management, and input/output handling. If the operating system has efficient input and output procedures, then STATJOB will not have to include special provisions to handle variable length records. The efficient storage and retrieval of intermediate computations or data is also dependent on equipment and the operating system, and hopefully will not need to be part of STATJOB.

It is expected that future interactive - remote access use of STATJOB will require only slight changes to STATJOB because of its original design. The routines that will need major modification to make STATJOB operate interactively are the STATJOB interface with the operating system and the control card processor. Essentially these routines, see Figure V. 3. 1, will need to be modified to operate as incremental interface controllers. In turn, these incremental interface controllers may require special routines in the input processor and the output processor, depending upon the operating system. It is expected that the memory management routines will have to be modified but we expect no major difficulties here.

At this time, we have working on the UWCC Time Sharing System on the Burroughs B5500, a prototype program for the interactive conversational generation of control cards for the 1604 Batch version of STATJOB. The purpose of this program is to learn how to have a user select on an interactive conversational basis the options from the various STATJOB capabilities. We already know we need a teaching mode and a production mode. We believe we will also need spelling variation routines such as found in PLANIT, see Feingold and Frye (31), and to a lesser extent in WIPL (Wisconsin Interactive Problem Solving Language), see Harris and Janowski (39).

We continue to look upon STATJOB as a production system and a vehicle to learn how to do better statistical data processing. Consequently, we expect to continue to determine additional ways and areas to achieve

standardization and subsequent sharing of the sub-components of STATJOB because, based upon our experience to date, we believe such standardization has the positive effects of providing the user better, easy to use computer capability for statistical data analysis. It also helps provide us with insights into what is needed in the STATJOB system.

Improvements in the analytical power of STATJOB will depend on the availability of better theory and algorithms for data analysis. If they become available, we believe the modular design of STATJOB will permit them to be directly integrated into it. The ability to extend STATJOB to include more extensive iterative types of data analysis than are now available will depend on overcoming some limitations to be summarized in the next Section.

VI. THE FUTURE

In looking to the future and trying to anticipate what it holds, it is easy to imagine that computers will be increasingly important instruments in data analysis because of the amazing accomplishments with computers in terms of their general availability, power, reliability, flexibility, and price/performance. It is easy to imagine that major improvements can be expected in using computers to acquire data, to screen and verify data, and to make the storage and retrieval of data easier and more economical so as to provide more opportunities for data analysis. In some areas of data analysis, computers have already been able to make significant contributions. They have been able to help provide desired focusing power needed for microscopic analysis, such as making it possible to do elaborate transformations of data to aid analysis. They have been able to help provide power needed for handling massive amounts of data and performing extensive calculations. However, in both the small and the large the power of the computer has also been surprisingly ineffective as an instrument for data analysis. In looking at the future, it is important

to understand why. Part of the explanation can be based upon the belief that much of data analysis involves iterative processing using non-quantitative insight and an artistic flair for looking at data. Part of the explanation can be found by taking into account that the individual with a data analysis problem has been moved away from his data when using a computer. As noted in earlier sections, the individual has been moved away from the computer and his data in an attempt to maximize the use of the equipment. I believe that by keeping in mind that today there are significant gaps between aspirations and realizations with regard to digital computers as instruments for data analysis, especially in the data rich situations, there is great hope for improvements.

Currently there is much enthusiasm and hope that interactive time-sharing and graphical input and output capabilities will be the basis for bridging the gaps between aspirations and realizations because they will aid iterative and visual data analysis. There is little, if any, doubt in my mind that these capabilities will enhance the role of the computer in data analysis, but I also believe there are some fundamental obstacles which need significant attention in the coming years. One is the need for better analytical tools which are iterative and sensitive to the amount of data available. It seems to me that the entire area of interactive time-sharing data analysis is in the need of analytical techniques which can work on data in an incremental manner so that all of the data do not need to be in the computer prior to starting the analysis. If this change can be realized, then economical interactive time-sharing computing ought to take place on a large scale.

When one is in the data poor situation, I do not see any fundamental limits imposed on the analysis by the computer. In the data rich situation I believe we have definite limits imposed upon us by current equipment, algorithms, and programming.

In the data rich situation, I believe that current technology is far too expensive and unreliable to make it possible for many to afford interactive time-sharing data analysis, but this situation need not hold true in the next ten years if adequate attention is given to analytical aspects of this problem. Although much interest and effort is being spent in establishing large data bases, I question for the reasons cited in Section IV that such files can be used on an interactive basis at this time. In addition to needing more reliable and efficient equipment, much more attention needs to be given not only to effective tools for storage and retrieval, but also to effective tools for maintenance of large data bases. This is an area which statisticians have not considered very much. Large data base problems, in my opinion, need analytical techniques which aid identification, description, and summarization.

In spite of the promise of extensive interactive time-sharing, there are many situations where a different approach is needed. I believe that since data analysis is an iterative process, we need periods for reflective thinking about intermediate results. Thus, I believe that Tukey's suggested approach of break-integrated processing, see (84), is what is needed in the future and will be recognized in the future. I also believe, as noted at the end of the Introduction, that time-sharing systems cannot relieve the user from needing extensive documentation so that he can study and understand the basis of the analysis. However, computer aided instruction in the future could help one to learn the documentation, see for example Schatzoff (74).

We should expect to see major improvements in the quality and price/performance of graphics devices with a subsequent major impact on data analysis.

We should expect to see continued effort spent on developing languages for Statistical Data Processing, especially in view of recent interest in

Statistical Languages, see for example, Bargmann (3), Claringbold (17), Miller (57), and Schatzoff (74). In 1959 I thought that this language development was vitally needed, (62). However, I now believe that such language capability will be needed primarily by the less sophisticated users or occasional users of computers because of improvements in statistical data processing capabilities which can be made available through advanced versions of systems of the type represented by STATJOB.*

It has been my recent experience that beginners or occasional users desire highly interactive and simple to use language approaches to computing but that the experienced users sophisticated in statistics and computers want shortcut or abbreviated ways of selecting options available from packaged systems. This need for compactification so as to avoid the requirement of always using a lengthy conversational approach, I believe will become more clear with the resulting expectation that two modes of use will develop as a standard part of the design of the language or statistical system. Thus I expect that extensions to existing statistical systems will take place. In some applications it seems reasonable to imagine that a Statistical Language will be used to specify the problem and the data and that this information will be analyzed, processed, and stored so that it can be subsequently requested through either a conversational approach or a brief set of identifiers or codes.

With regard to the future of data analysis and the computer, I am optimistic about the possibility of improvement but cautious about the time scale of implementation. I hope and believe that greater use will be made of the computer to select samples and to use data on a sampling basis. I hope that the use of computers will be expanded in the Administrative Control function cited in Section I for I believe this is an area where computers can make a major contribution to the acquisition and control of data. This is an area where I hope and believe that techniques

*As noted by John W. Tukey, some people may consider STATJOB a statistical language.

and programs will be developed to aid in the interactive design and generation of statistical designs and the subsequent collection of the data. These developments could and should significantly expand the role of the computer as an instrument for data analysis in the near future. With additional analytical tools, I expect that on-line data acquisition will continue to increase and be a great help to improving man's insight and productivity.

VII. SUMMARY

Attention was focused on the need to look at statistical data analysis involving computers as a multi-dimensional problem so as to gain better insight and understanding. Attention was then shifted to a review of the use of computers in statistical analysis. The problems associated with large data files were considered. This was followed by a summary description of the objectives and status of STATJOB. The future was then considered in terms of some of the things that need to be done.

In conclusion then, based upon successes of the past 20 years, it seems reasonable to expect better and more economical equipment in the years ahead. This, in turn, will make it possible to expand the fields of use of computers as instruments for data analysis. As equipment for storage and retrieval of data improves, it seems reasonable to expect associated improvements in programming. The challenge for some statisticians will be to participate in these developments and the determination of the required statistical techniques. While the gaps between aspirations and realizations are great (especially in the data rich applications), if we are willing to admit these gaps exist and if we are willing to devote time and effort to filling in some of these gaps, then the future of computers as instruments for data analysis is indeed bright.

LIST OF REFERENCES

1. Anscombe, F. J. (1967). "Topics in the Investigation of Linear Relations Fitted by the Method of Least Squares (with Discussion)", Jour. Roy. Stat. Soc., Series B, Vol. 29, No. 1, pp 1-52

2. Anscombe, F. J., and Tukey, J. W. (1963). "The Examination and Analysis of Residuals", Technometrics, Vol. 5, p. 141-160

3. Bargman, R. E. (1965). "A Statistician's Instructions to the Computer: A Report on a Statistical Computer Language", IBM Scientific Computing Symposium - Statistics, pp. 301-323

4. Barrodale, I. (1968) "L_1 Approximation and the Analysis of Data", Applied Statist., Vol. 17, pp. 51-57

5. Baur, F. L. (1965). "Elimination with Weighted Row Combinations and Least Squares Problems", Numerische Mathematik, Vol. 7, pp. 338-352

6. Beaton, A. E. (1964). "The Use of Special Matrix Operations in Statistical Calculus", Research Bulletin, Educational Testing Service, Princeton, N. J.

7. Björck, A. (1967). "Solving Linear Least Squares Problems by Gram-Schmidt Orthogonalization", Nordisk Tidskr. Information-Behandling, Vol. 7, pp. 1-21

8. Bock, R. D. (1963). "Programming Univariate and Multivariate Analysis of Variance", Technometrics, Vol. 5, pp. 95-117

9. Bock, R. D. (1965). "A Computer Program for Univariate and Multivariate Analysis of Variance", Proceedings IBM Scientific Computing Symposium-Statistics, pp. 69-111, IBM

10. Booth, G. W., Box, G. E. P., Muller, M. E., and Peterson, T. I. (1959). "Forecasting by Generalized Regression Methods, Nonlinear Estimation, (Princeton-IBM)", International Business Machines Corporation, New York

11. Box, G. E. P., and Coutie, G. A. (1956). "Application of Digital Computers in the Exploration of Functional Relationships", Proc. Inst. Elec. Engrs., Vol. 103, Part B, Supplement No. 1, pp. 100-107

12. Box, G. E. P., and Cox, D. R. (1964). "An Analysis of Transformations", Jour. Roy. Stat. Soc., Vol. 26, pp. 211-243

13. Buhler, R. (1966). "P-STAT-An Evolving User-oriented Language for Statistical Analysis of Social Science Data", Computer Center, Princeton University, Version 43

14. Bullard, E. C., Oglebay, F. E., Munk, W. H., and Miller. G. R. (1964). "A User's Guide to BOMM, A System of Programs for the Analysis of Time Series", Institute of Geophysics and Planetary Physics, University of California, La Jolla

15. Chambers, J. M. (1967). "Some General Aspects of Statistical Computing", Applied Statist., Vol. 16, No. 2, pp. 124-132

16. Champomier, L. (1964). "SUMX", Lawrence Radiation Laboratory, University of California, Berkeley, UCRL-1122

17. Claringbold, P. J. (1967). "Towards SPL, A Statistical Programming Language", Proceedings VI International Biometric Conference, pp. 6.37-6.52, Sydney, Australia, August

18. Clyde, D. J., Cramer, E. M., Sherin, R. J. (1966). "Multivariate Statistical Programs", Biometric Laboratory, University of Miami

19. Colin, A. J. T. (1967). "On-line Access Systems in Statistics", Applied Statist., Vol. 16, No. 2, pp. 111-119

20. Cooley, J. W. and Tukey, J. W. (1965). "An Algorithm for the Machine Calculation of Complex Fourier Series", Math. of Computation, Vol. 19, No. 90, pp. 297-301

21. Cooley, W. W. and Lohnes, P. R. (1962). Multivariate Procedures for The Behavioral Sciences, J. Wiley

22. Cooper, B. E. (1967). "ASCOP-A Statistical Computing Procedure", Applied Statist., Vol. 16, No. 2, pp. 100-110

23. Cooper, B. E (1967). "A Unifying Computational Method for the Analysis of Complete Factorial Experiments, Comm. A. C. M., Vol. 10, pp. 27-34

24. Craddock, J. M. and Freeman, M. H. (1967). "The METO Computer Language", Applied Statist., Vol. 16, pp. 120-122

25. Daniel, C. (1959). "Use of Half-Normal Plots in Interpreting Factorial Two-level Experiments", Technometrics, Vol. 1, pp. 311-342

26. Dixon, W. J. (1967). BMD "Biometrical Computer Programs Health Science Computation Facility", University of California, Los Angeles

27. Dixon, W. J. (1967). "Use of Displays with Packaged Statistical Programmes", Proceedings VI International Biometric Conference, pp. 6.89-6.99, Sydney, Australia, August

28. Dodson, G., Parker, L., and Rosbrook, G. (1964). "SSUPAC", Manual of Computer Programs for Statistical Analysis, Statistical Service Unit, University of Illinois, Urbana

29. Fan, C. T., Muller, M. E., and Rezucha, I. (1962). "Development of Sampling Plans by Using Sequential (Item by Item) Selection Techniques and Digital Computers", J. Amer. Statist. Assoc., Vol. 57, pp. 387-402

30. Fan, C. T., and Scheidemantle, R. (1964). "ISSS-Information Selection and Sampling System for IBM 7090/94", SHARE General Program Library, 7090/94 SO IBM 0025

31. Feingold, S. L., and Frye, C. H. (1966). "User's Guide to PLANIT-Programming Language for Interactive Teaching", Tech. Memo. 3055/000/01, pp. 1-214, System Development Corporation, November

32. Forsythe, G. E., and Moler, C. B. (1967). Computer Solution of Linear Algebraic Systems, Prentice-Hall, Englewood

33. Gabor, J. B., and Carlitz, R. D (1965). "TSAR - Tape Storage and Retrieval System; User's Manual", Duke University Computing Laboratory

34. Gentleman, W. M. (1968). "Further Comment on the Computation of Least Squares Polynomials - Letter to Editor", Comm. A. C. M., Vol. 11, p. 73

35. Goldberg, M. (1967). "On the Computation of Least Squares Polynomials", Comm. A. C. M., Vol. 10, pp. 56-57

36. Golub, G. (1965). "Numerical Methods for Solving Linear Least Squares Problems", Numerische Mathematik, Vol. 7, pp. 206-216

37. Golub, G. H., and Wilkinson, J. H. (1966). "Note on the Iterative Refinement of Least Squares Solution", Numerische Mathematik, Vol. 9, pp. 139-148

38. Gower, J. C., Simpson, H. R., and Martin, A. (1967). "A Statistical Programming Language", Applied Statist., Vol. 16, No. 2, pp. 89-99

39. Harris, E., and Janowski, R. M. (1968). "WIPL - User Manual (#1000). Draft 1", University of Wisconsin Computing Center

40. Hartley, H. O. (1956). "A Plan for Programming Analysis of Variance for General Purpose Computers", Biometrics, Vol. 12, pp. 110-122

41. Hastings, C. (1955). Approximations for Digital Computers, Princeton University Press

42. Healy, M. J. R. (1967). "Invited Contributions to Meeting on Statistical Programming", Applied Statist., Vol. 16, No. 2, pp. 135-136

43. Healy, M. J. R., and Westmacott, M. H. (1956). "Missing Values in Experiments Analyzed on Automatic Computers", Applied Statist., Vol. 5, pp 203-206

44. Hilsenrath, J., Ziegler, Guy C., Messiwa, C. G., Walsh, P. J., and Herbold, R. J. (1966). "OMNITAB, A Computer Program for Statistical and Numerical Analysis", National Bureau of Standards, Handbook 101

45. Hutchins, C. W., Wetterstrand, W. H., and Wolfe, P. J. (1966). "STATJOB", University of Wisconsin Computing Center, Vol. VI

46. Hutchins, C. W., Muller, M. E., Wolfe, P. J. "A Transformation Language Capability for Statistical Data Analysis", University of Wisconsin Computing Center, In Preparation

47. IBM. (1967). "Numerical Surface Techniques and Contour Map Plotting", Data Processing Division, IBM E20-0117-0

48. IBM. (1963). "STORM - Statistically Oriented Matrix Programming System", Federal System Division IBM, Bethesda, Maryland

49. IBM. (1966). "System/360 Scientific Subroutine Package", (360A-CM-03X), Programmer's Manual, H20-0205-0, IBM, White Plains

50. Jansson, B. (1966). Random Number Generators, Almquist and Wiksell, Stockholm

51. Jones, H. L. (1959). "How Many of a Group of Random Numbers Will be Usable in Selecting a Particular Sample?", Jour. Amer. Statist. Assoc., Vol. 54, pp. 102-122

52. Jordan, T. L. (1968). "Experiments on Error Growth Associated With Some Linear Least-Squares Procedures", Math. Comp., Vol. 22, pp. 579-588

53. Kronmal, G. A., and Yarnall, S. R. (1967). "A Conversational Mode Computer System for Statistical Analysis", Proceedings VI International Biometric Conference, pp. 6.66-6.79, Sydney, Australia, August

54. Leone, F. C. (1961). Abstract of Statistical Computer Routines, Case Institute of Technology, 225 pages, November

55. Lewis, P. A. W. (1966). "A Computer Program for the Statistical Analysis of Events", IBM Journal, Vol. 5, pp. 202-225

56. Lindley, D. V. (1968). "The Choice of Variables in Multiple Regression (With Discussion)", J. Roy. Stat. Soc., Vol. 30, pp. 31-66

57. Miller, J. R. (1967). "DATANAL-An Interpretive Language for On-Line Analysis of Empirical Data", Working Paper - Sloan School MIT, #275-67, August

58. Moulton, P. G., and Muller, M. E. (1967). "DITRAN - A Compiler Emphasizing Diagnostics", Comm. A. C. M., Vol. 10, No. 1, pp. 45-52, January

59. Muller, M. E. (1958). "The Use of Computers in Inspection Procedures", Comm. A. C. M., Vol. 1, pp. 7-13

60. Muller, M. E. (1959). "The Role of Existing Computer Techniques in Statistics and Quality Control", Industrial Quality Control, Vol. XV, No. 9, pp. 2-5

61. Muller, M. E. (1962). "Techniques and Programming for Sampling Populations on Magnetic Tapes, With a Summary of the Results by C. T. Fan, M. E. Muller, and I. Rezucha", Transactions of the Middle-Atlantic Conference of A. S. Q. C., pp. 249-285, March

62. Muller, M. E. (1962). "Impact and Use of Computers in Quality Control", Transactions, A.S.Q.C., pp. 225-230

63. Muller, M. E. (1965). "Computing Techniques for Cluster Sampling", Proceedings IBM Scientific Computing Symposium-Statistics, pp. 169-184, IBM

64. Muller, M. E. (1968). "Random Numbers", International Encyclopedia of the Social Sciences, MacMillian, pp 307-313

65. Muller, M. E., Wetterstrand, W., and Wolfe, P. J. "The STATJOB System", In Preparation, University of Wisconsin Computing Center

66. National Bureau of Standards. (1964). Handbook of Mathematical Functions With Formulas, Graphs, and Mathematical Tables, edited by Abromowitz, M., and Stegun, I. A., Applied Mathematics Series 55

67. Neely, P. M. (1966). "Comparison of Several Algorithms for Computation of Means, Standard Deviations, and Correlations Coefficients", Comm. A.C.M., Vol. 9, pp. 496-499

68. Nelder, J. A. (1966). "General Statistical Program (GENSTAT Version IV), User's Guide, Biometry Section, Waite Institute, Glen Osmond, South Australia

69. Nelder, J. A. (1965). "The Analysis of Randomised Experiments with Orthogonal Block Structure, I. Block Structure and the Null Analysis of Variance", Proc. Roy. Soc. A, Vol. 283, pp. 147-162

70. Nelder, J. A. (1965). "The Analysis of Randomised Experiments with Orthogonal Block Structure, II. Treatment Structure and the General Analysis of Variance", Proc. Roy. Soc. A, Vol. 283, pp. 163-178

71. O'Brien, W. (1967). "COSMOS-Courtauld's Own System for Matrix Operations and Statistics", Computer Journal, Vol. 10, pp. 128-134

72. Oliver, I. (1967). "Analysis of Factorial Experiments Using Generalized Matrix Operations", Jour. A.C.M., Vol. 14, pp. 508-519

73. Rodden, B.E. (1967). "Error-Free Methods for Statistical Computation", Comm. A.C.M., Vol. 10, pp. 179-180

74. Schatzoff, M. (1965). "Console Oriented Model Building", Proceedings, 20th National Conference, A. C. M., pp. 354-373

75. Schlater, J. E., and Hemmerle, W.J. (1966). "Statistical Computations Based Upon Algebraically Specified Models", Comm A. C. M., Vol. 9, pp. 865-869

76. Shannon, S., and Henschke, C. (1967). "Stat-Pack: A Biostatistical Programming Package", Goodard Computer Science Institute, Comm. A. C. M., Vol. 10, pp. 123-125

77. Singleton, R. C. (1967). "On Computing the Fast Fourier Transform", Comm. A. C. M., Vol. 10, pp. 647-654

78. Stacy, R. W., and Waxman, B. D. (1965). Computers in Biomedical Research, Vol. I and II, Academic Press, in particular the article by W. Siler, "Hybrid Computers in Bioscience", Chapter 5, Vol. I.

79. Sterling, T., and Pollack, S. (1964). "MEDCOMP-Statistical Systems", Medical Computing Center, University of Cincinnati, Ohio Revised Handbook

80. Tauchi, H. J. (1965). "The Use of QUICKTRAN as a Problem-Oriented Language", presented at Fourth Annual Symposium on Biomathematics and Computers in the Life Sciences, Houston, Texas, March

81. Tauchi, H. J. (1965). "Abstract of Statistical Computer Programs", SHARE Statistical Program Evaluation Committee, January

82. Taussky, O. (1950). "Note on the Condition of Matrices", Math. Tables Aids Comput., Vol. 4, pp. 111-112

83. Tukey, J. W. (1962). "The Future of Data Analysis", Ann. of Math. Statist., Vol. 33, pp. 1-67

84. Tukey, J. W. (1965). "The Inevitable Collision Between Computation and Data Analysis", Proceedings IBM Scientific Computing Symposium - Statistics, pp. 141-152, IBM

85. Tukey, J. W., and Wilk, M. B. (1966). "Data Analysis and Statistics: An Expository Overview", Proceedings-Fall Joint Computer Conference, pp. 695-709

86. UNIVAC. (1965). "UNIVAC 1108 STAT-PACK Program Abstracts", UP-4041 Sperry Rand Corp.

87. Van Reeken, A. J. (1968). "Dealing With Neely's Algorithms", Comm. A. C. M., Vol. 11, pp. 149-150

88. Wilk, M. B., and Gnanadesikan, R. (1968). "Probability Plotting Methods for the Analysis of Data", Biometrika, Vol. 55, pp. 1-17

89. Wilkinson, G. N. (1967). "A General Recursive Algorithm for Analysis of Variance", Proceedings VI International Biometric Conference, Sydney, Australia, August, pp. 6.13 - 6.36

90. Wilkinson, J. H. (1963). Rounding Errors in Algebraic Processes, Prentice-Hall

91. Wolfe, P. J. (1966). "Non-Linear Least Squares (GAUSHAUS), University of Wisconsin Computing Center, Vol. IV, Rev. B, pp. 1-41

92. Wolontis, V. M. (1956). "A Complete Floating-Decimal Interpretive System for the IBM 650 Magnetic Drum Calculator", Technical Newsletter No. 11, Applied Science Division, IBM, March

93. Yates, F., and Anderson, A. J. B. (1966). "A General Computer Programme for the Analysis of Factorial Experiments", Biometrics, Vol. 22, pp. 503-524

Acknowledgements

The author wishes to express his appreciation to Peter J. Wolfe for his helpful comments on the first draft of the manuscript. The careful reading of the paper and the subsequent useful suggestions received from Dr. John W. Tukey are also gratefully acknowledged as is the encouragement received from Dr. George E. P. Box, Dr. M. D. Godfrey, Dr. Irwin Guttman, and William Wetterstrand.